

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平8-44630

(43) 公開日 平成8年(1996)2月16日

(51) Int.Cl.⁶

G 0 6 F 12/14
12/00

識別記号

3 1 0 K

庁内整理番号

5 3 7 A 7623-5B

F I

技術表示箇所

審査請求 未請求 請求項の数 4 O L (全 9 頁)

(21) 出願番号 特願平6-182575

(22) 出願日 平成6年(1994)8月3日

(71) 出願人 000155469

株式会社野村総合研究所

東京都中央区日本橋1丁目10番1号

(72) 発明者 真下 竜 実

神奈川県横浜市保土ヶ谷区神戸町134番地

株式会社野村総合研究所内

(72) 発明者 小野 喜代志

神奈川県横浜市保土ヶ谷区神戸町134番地

株式会社野村総合研究所内

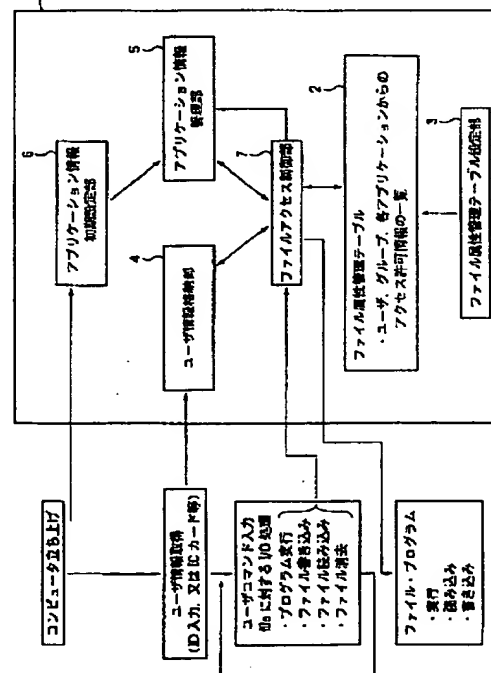
(74) 代理人 弁理士 佐藤 一雄 (外3名)

(54) 【発明の名称】 ファイルアクセス制御装置およびその制御方法

(57) 【要約】

【目的】 複数のアプリケーションソフトウェアからなるソフトウェアの誤作動や不具合あるプログラムの実行による広範なデータ破壊を効果的に防止するファイルアクセス制御装置およびその制御方法を提供する。

【構成】 ファイルごとにそれにアクセスできるユーザとアプリケーションソフトウェアとそのファイルに対する処理の種類を定義したファイル属性管理テーブル2と、ソフトウェアを実行しているユーザの情報を格納するユーザ情報格納手段4と、実行されているアプリケーションソフトウェアの情報を管理するアプリケーション情報管理手段5と、ユーザ情報格納手段4とアプリケーション情報管理手段5とファイル属性管理テーブル2とを参照してファイルに対するアクセス命令を許可あるいは禁止するファイルアクセス制御手段7とを備えた。



1

【特許請求の範囲】

【請求項1】ソフトウェアを構成するファイルごとにこれにアクセスできるユーザとアプリケーションソフトウェアとそのファイルに対する処理の種類を定義したファイル属性管理テーブルと、

前記ソフトウェアを実行中のユーザの情報を格納するユーザ情報格納手段と、

実行中のアプリケーションソフトウェアの情報を管理するアプリケーション情報管理手段と、

前記ファイルに対するアクセス命令が発せられたときに、前記ユーザ情報格納手段と前記アプリケーション情報管理手段とを参照して前記アクセス命令を発したユーザおよびアプリケーションソフトウェアを検出し、前記ファイル属性管理テーブルの定義内容によって前記ファイルに対するアクセス命令を許可あるいは禁止するファイルアクセス制御手段とを備えたことを特徴とするファイルアクセス制御装置。

【請求項2】ファイルごとにこれを読み出し、書き込み、実行および消去できるユーザとアプリケーションソフトウェアとをコンピュータの画面上で設定可能なファイル属性管理テーブル設定手段を備えたことを特徴とする請求項1に記載のファイルアクセス制御装置。

【請求項3】ソフトウェアを構成するファイルごとにこれを読み出し、書き込み、実行および消去することができるユーザとアプリケーションソフトウェアを定義してコンピュータの記憶装置に格納し、

前記ソフトウェアを実行するユーザにユーザ情報を入力させてそのソフトウェアを実行しているユーザとして登録し、

ファイルに対するアクセスの状態を把握することによって、現に実行されているアプリケーションソフトウェアを登録し、

ソフトウェアをファイルに対するアクセス命令が発せられたときに、そのアクセス命令を発したユーザとアプリケーションソフトウェアを検出し、前記ファイルにアクセス可能なユーザとアプリケーションソフトウェアの定義に照らしてそのアクセス命令を許可あるいは禁止することを特徴とするファイルアクセス制御方法。

【請求項4】ソフトウェアが使用される条件に応じて、コンピュータの画面上でソフトウェアを構成するファイルごとにこれを読み出し、書き込み、実行および消去することができるユーザとアプリケーションソフトウェアを定義することを特徴とする請求項3に記載のファイルアクセス制御方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、プログラムやデータのファイルに対するアクセスを制御することにより、誤作動や不正なプログラムの実行によるプログラムやデータの破壊や、ウィルスの感染による大規模なデータ破壊を

2

防止する装置および方法に係り、特にプログラムやデータのファイルごとにこれにアクセス可能なユーザやプログラム（以下、アプリケーションソフトウェアあるいは単にアプリケーションという）を定義し、これによってアプリケーションの連鎖的な実行による大規模なデータ破壊を防止するファイルアクセス制御装置およびその制御方法に関する。

【0002】

【従来の技術】一般にソフトウェアは、一つのソフトウェアの内部に複数のプログラムやデータのファイルを含み、これらプログラム（アプリケーション）を切り替えてながら実行してゆくようにしている。

【0003】このようなものとしては、たとえばゲーム用ソフトウェアがある。ゲーム用ソフトウェアでは、最初の画面から次々に新しい画面が展開され、展開された新しい画面ではその画面に沿った動作を行って全体として一つのゲームを楽しめるように構成されている。このような手法はゲーム用ソフトウェアに限らず、現在各種のソフトウェアに広く用いられている。

【0004】上記のようなソフトウェアでは、アプリケーション同士が複雑に関連し合い、一つのアプリケーションが複数の他のアプリケーションによって読み出されあるいは実行されることが可能であり、場合によってはプログラムやデータのファイルが他のアプリケーションによって書き換えあるいは消去されることがある。

【0005】このような構成のソフトウェアでは、オペレーションミスにより、あるいは不具合な部分を有するアプリケーションの実行により、無関係なプログラムやデータのファイル（本来そのアプリケーションの実行によってはアクセスされないファイル）が破壊されたり、さらに連鎖的に無関係なファイルが破壊されたりして、全体的に深刻なデータ破壊を招く危険性があった。

【0006】特に最近問題になっているウィルス感染は、上記現象を意図的に狙ったものであり、データ破壊を隠蔽したプログラムを実行させることにより、連鎖的に広範なデータ破壊を生じさせるものである。

【0007】これに対して従来からアプリケーションによってプログラムやデータのファイルに対するアクセスを制御する装置および方法が提案されていた。この従来のファイルアクセス制御装置および方法は、書き込み不可能なプログラムやデータと書き込み可能なプログラムやデータをそれぞれ、メモリ上のアクセス禁止領域とアクセス許可領域にロードし、アクセス命令の発信元やアクセスする先のプログラムやデータをメモリ上のアドレスによって管理し、これによってファイルに対するアクセスを制御するものであった（特開昭57-71597号公報参照）。

【0008】一方、ユーザによってプログラムやデータのファイルに対するアクセスを許可あるいは禁止する装置および方法が従来からあった。これは、正規の使用を

10

20

30

40

50

3

しようとするユーザにのみファイルに対するアクセス権を与えて、ソフトウェアを保護しようとするものであった。具体的には、ユーザの識別情報を確認することによってユーザ単位で、あるいはユーザグループ単位でファイルに対するアクセスを許可あるいは禁止するものであった。

【0009】

【発明が解決しようとする課題】しかしながら、上記プログラムやデータのファイルをメモリのアクセス可能領域とアクセス禁止領域にロードしてメモリ上のアドレスによって管理する従来の方法では、ソフトウェアの構成に対する制限が多かった。すなわち、複雑なソフトウェアの場合、あるプログラムやデータのファイルは所定のプログラム等からのアクセスに対しては書き込み禁止のファイルとなり、他のプログラム等からのアクセスに対しては書き込み可能なファイルとなるようにきめ細かく構成する必要がある。これに対して、メモリ上の領域によって書き込み可能、書き込み禁止を区別する上記従来の方法では、このような細かい設定を行うことができなかった。

【0010】また、一旦設定した書き込み可能、書き込み禁止の設定をソフトウェアの実行条件によって柔軟に変えることもできなかった。さらに、ユーザによって所定のファイルへのアクセスを許可あるいは禁止することも不可能であった。

【0011】一方、上記ユーザの認証情報を確認することによってアプリケーションに対するアクセスを許可あるいは禁止する従来の方法は、一旦ユーザが正規のユーザと認められれば、そのユーザはそのソフトウェア中のいかなるファイルへのアクセスも可能となる。この結果、正規のユーザの実行命令がオペレーションミスであったり、実行されるプログラムに不具合がある場合には、アプリケーションの連鎖的なデータ破壊を生じることがあった。

【0012】特に、ウィルス感染のソフトウェアは、不具合のあるプログラムを隠蔽し、これを正規のユーザに実行させるので、上記従来の方法では、ウィルスによる広範なデータ破壊を防止することができなかった。

【0013】そこで、本発明の目的は、上記従来の技術の課題を解決し、複数のアプリケーションソフトウェアからなるソフトウェアの誤作動や不具合あるプログラムの実行による広範なデータ破壊を効果的に防止するファイルアクセス制御装置およびその制御方法を提供することにある。

【0014】

【課題を解決するための手段】上記目的達成のために、本願の請求項1に係るファイルアクセス制御装置は、ソフトウェアを構成するファイルごとにそれにアクセスできるユーザとアプリケーションソフトウェアとそのファイルに対する処理の種類を定義したファイル属性管理テ

4

ーブルと、前記ソフトウェアを実行中のユーザの情報を格納するユーザ情報格納手段と、実行中のアプリケーションソフトウェアの情報を管理するアプリケーション情報管理手段と、前記ファイルに対するアクセス命令が発せられたときに、前記ユーザ情報格納手段と前記アプリケーション情報管理手段とを参照して前記アクセス命令を発したユーザおよびアプリケーションソフトウェアを検出し、前記ファイル属性管理テーブルの定義内容によって前記ファイルに対するアクセス命令を許可あるいは禁止するファイルアクセス制御手段とを備えたことを特徴とするものである。

【0015】本願の請求項2にかかるファイルアクセス制御装置は、上記請求項1のファイルアクセス制御装置において、ファイルごとにこれを読み出し、書き込み、実行および消去できるユーザとアプリケーションソフトウェアとをコンピュータの画面上で設定可能なファイル属性管理テーブル設定手段を備えたことを特徴とするものである。

【0016】本願の請求項3にかかるファイルアクセス制御方法は、ソフトウェアを構成するファイルごとにこれを読み出し、書き込み、実行および消去することができるユーザとアプリケーションソフトウェアを定義してコンピュータの記憶装置に格納し、前記ソフトウェアを実行するユーザにユーザ情報を入力させてそのソフトウェアを実行しているユーザとして登録し、ファイルに対するアクセスの状態を把握することによって、現に実行されているアプリケーションソフトウェアを登録し、ソフトウェアをファイルに対するアクセス命令が発せられたときに、そのアクセス命令を発したユーザとアプリケーションソフトウェアを検出し、前記ファイルにアクセス可能なユーザとアプリケーションソフトウェアの定義に照らしてそのアクセス命令を許可あるいは禁止することを特徴とするものである。

【0017】本願の請求項4に係るファイルアクセス制御方法は、上記請求項3のファイルアクセス制御方法において、ソフトウェアが使用される条件に応じて、コンピュータの画面上でソフトウェアを構成するファイルごとにこれを読み出し、書き込み、実行および消去することができるユーザとアプリケーションソフトウェアを定義することを特徴とするものである。

【0018】

【作用】本願の請求項1および請求項3のファイルアクセス制御装置および制御方法は、ソフトウェアを構成するファイルごとに、それにアクセス可能なユーザとアプリケーションとそのファイルに対する処理の種類を予めファイル属性管理テーブルに定義・登録し、ユーザ情報格納手段にソフトウェアを実行しているユーザの情報を格納し、アプリケーション情報管理手段に現に実行されているアプリケーションソフトウェアの情報を格納し、ファイルに対するアクセス命令があったときに、上記ユ

10

20

30

40

50

ユーザ情報格納手段とアプリケーション情報管理手段からそのアクセス命令を発したユーザとアプリケーションを検出し、上記ファイル属性管理テーブルに照らしてそのアクセス命令を許可あるいは禁止することができる。

【0019】これにより、上記本発明のファイルアクセス制御装置および制御方法は、ソフトウェアを構成する各ファイルに対してアクセスできるユーザ、アプリケーション、およびそのファイルに対する処理の種類（たとえば、読み出し、書き込み、実行、消去等）を特定でき、ソフトウェアを構成するアプリケーションの正しい作動による処理を保証するとともに、誤作動や不具合なプログラムの実行によるファイルの連鎖的なデータ破壊を防止することができる。

【0020】また、本願の請求項2および請求項4のファイルアクセス制御装置および制御方法では、コンピュータの画面上でソフトウェアを構成するファイルごとにこれを読み出し、書き込み、実行および消去することができるユーザとアプリケーションソフトウェアを定義することができるので、ソフトウェアが使用される条件に応じて、各ファイルに対するアクセス条件を随時変更することができる。また、このようなファイルに対するアクセス条件を任意に定義できる自由さにより、種々のソフトウェアに対して適用可能なファイルアクセス制御装置および制御方法を得ることができる。

【0021】

【実施例】以下本発明によるファイルアクセス制御装置及びその制御方法の一実施例について添付の図面を用いて説明する。

【0022】図1は、本発明によるファイルアクセス制御装置の一実施例の構成とその処理の流れを示している。

【0023】本実施例のファイルアクセス制御装置1は、ソフトウェアを実行するコンピュータと別のハードウェアによっても構成することができるが、好ましくはソフトウェアによって各ソフトウェアを実行するコンピュータの処理装置、記憶装置、入出力装置等を利用して構成する。

【0024】図1に示すように、本実施例のファイルアクセス制御装置1は、実行しようとするソフトウェアのファイルごとに、それにアクセスすることができるユーザ、ユーザグループ、アプリケーションソフトウェアを定義したファイル属性管理テーブル2と、前記ファイル属性管理テーブル2の内容を設定するファイル属性管理テーブル設定部3と、現在ソフトウェアを実行しているユーザに関する情報を格納するユーザ情報格納部4と、現在実行中のコマンドがどのアプリケーションソフトウェアのコマンドであるかを把握するアプリケーション情報管理部5と、前記アプリケーション情報管理部5と、前記アプリケーション情報管理部5の初期設定を行うアプリケーション情報初期設定部6と、ファイルに対する

アクセス命令があったときに前記ユーザ情報格納部4とアプリケーション情報管理部5とファイル属性管理テーブル2とを参照してファイルに対するアクセスを許可あるいは禁止するファイルアクセス制御部7とからなる。

【0025】次に、上記構成に基づくソフトウェア実行中の上記ファイルアクセス制御装置1の作用について説明する。

【0026】すでにファイル属性管理テーブル2が設定されているとして、最初にコンピュータを立ち上げた時にアプリケーション情報初期設定部6にコンピュータ起動の信号が入力され、これによってアプリケーション情報が初期状態に設定される。通常、このアプリケーション情報初期設定部6の作用によって、アプリケーション情報管理部5はアプリケーションソフトウェアが何も登録されていない状態に戻される。

【0027】コンピュータが立ち上がったところで、続いて直にユーザ情報の入力が必要される。これは、ソフトウェアを実行する前に、一次的に不正なユーザを排除するためである。

【0028】ユーザ情報を取得するには、ユーザにユーザIDの入力を求めたり、ICカードによる照合を求め等々の公知の方法によって行うことができる。

【0029】取得されたユーザ情報が正規のユーザのものであるときは、そのユーザによるソフトウェアの使用を許可し、得られたユーザ情報をユーザ情報格納部4に格納する。

【0030】このようにソフトウェアの使用を許可されたユーザによって、そのソフトウェアが実行されて種々の処理が行われる。これら処理には、ソフトウェアを構成する多数のファイル（プログラムのファイルと、データのファイルとがある）に対する入出力処理も含まれている。

【0031】本実施例のファイルアクセス制御装置1は、上記ファイルに対する入出力処理の命令は一旦ファイルアクセス制御部7へ入力され、アクセス制御部7によってそのアクセスを許可あるいは禁止する判断が行われ、しかる後に初めてファイルにアクセスできるように構成されている。このファイルに対する入出力処理の種類としては、プログラムの実行、ファイルへの書き込み、ファイルの読み出し、ファイルの消去がある。

【0032】ファイルアクセス制御部7は、ファイルに対する入出力処理の命令を受け取ったときは、その命令が新しいプログラムを読み込んで実行する命令であるときは、その新しいプログラム名をアプリケーション情報管理部5に登録する。これによって、常に実行中のアプリケーションに関する最新の情報を保持することができる。

【0033】一方、上記ファイルに対する入出力処理の命令が所定のファイルの内容を変更または消去する命令であるときは、ファイルアクセス制御部7は、ユーザ情

報格納部4とアプリケーション情報管理部5とファイル属性管理テーブル2の内容を参照し、その命令が、対象となっているファイルに対してアクセスする権限を与えられているユーザおよびアプリケーションソフトの命令であるか否か、また、その命令による処理の種類が許可されているものであるか否かを判断し、その命令を実行を許可あるいは禁止する。

【0034】次に上記判断の基準となるファイル属性管理テーブルの内容について説明する。

【0035】図2は、ファイル属性管理テーブル設定部3によるファイル属性管理テーブル2の内容設定の画面の一例を示している。

【0036】図2において、画面の上部は設定画面の検索条件を設定する部分であり、ファイル名、ユーザ名、ユーザグループ名、タイプ、ファイルIDの少なくとも1つを入力することにより、ファイル属性管理テーブル設定部3は、その検索条件に合致したファイルのファイル属性管理テーブルの設定画面を表示する。

【0037】上記設定画面検索条件を入力する部分の下方がファイル属性管理テーブルの内容を設定する画面である。

【0038】ファイル属性管理テーブルの内容を設定する画面の左側部分は、アクセスの対象となるファイルの名称（ここではmain.exe、image.dat、sub.exe、hiscore.dat）、アクセスを行なうユーザ（ α ）、ユーザグループ（ β ）、各ファイルに付与されるアプリケーションとしてのID（XXX）を設定する画面となっている。

【0039】上記ファイル、ユーザ等を設定する画面の右側には、各ファイルに対するアプリケーションのアクセス条件を設定する画面となっている。このアクセス条件設定画面は、大きくユーザ管理の設定画面とアプリケーション管理の設定画面に分けられている。

【0040】ユーザ管理の設定画面は、ユーザとユーザグループとユーザおよびユーザグループ以外のアクセス許可条件に分けられている。この区分けは例示であり、必要に応じてさらに細かい指定、たとえば各ユーザごとのアクセス許可条件を設定することもできる。

【0041】このユーザ管理の区分け欄の下方には、対象となるファイルに対する処理の種類（読み出し、書き込み、実行、消去）ごとに許可あるいは禁止する条件を設定する画面が表示されている。

【0042】同様に、アプリケーション管理の設定画面は、同一のアプリケーションIDからのアクセスと、異なるアプリケーションIDからのアクセスとに分けられ、それぞれのアプリケーションからのアクセスに対してその処理の種類ごとに許可あるいは禁止の条件を設定できるようになっている。ここでも、必要に応じてさらに細かい指定、例えばアプリケーションIDごとのアクセス条件を設定することができる。

【0043】図2の設定例では、ファイルmain.exeは、ユーザ・ユーザグループ以外の人には読み出しのみでき、正規のユーザ α とユーザグループ β は実行することができる。さらにユーザ α は、ファイルmain.exeを書き換えあるいは消去することができる。また、ファイルmain.exeは、全ての他のアプリケーション・ソフトウェアによって起動および実行されることが可能である。

【0044】これに対してファイルimage.datは、同一IDのアプリケーションのみによって参照および書き換えられ、しかもユーザ α のみが書き換えを行なうことができる。

【0045】また、ファイルsub.exeは、ユーザ α とユーザグループ β の人によって、main.exe（同一ID）を通じてのみ実行される。ユーザ α は、ファイルsub.exeを消去できるが、誤操作防止のために書き換えを行なうことはできない。

【0046】最後にファイルhiscore.datは、ユーザ α とユーザグループ β の人がファイルmain.exeまたはsub.exeを通じてのみ参照および書き換えを行うことができる。また、消去することができるのはユーザ α のみである。

【0047】このように本発明は、ファイル属性管理テーブル2のユーザ管理、アプリケーション管理および許可する処理の種類を設定することにより、ファイルごとにこれにアクセスできるユーザとアプリケーションと処理の種類を自由に設定することができる。また、上記ファイル属性管理テーブル2の設定は、ファイル属性管理テーブル設定部3によって必要な時に行うことができるので、ソフトウェアの実行条件、たとえばファイルにアクセスできるユーザを追加・変更したり、ソフトウェアの設計仕様の変更によって各ファイルにアクセスできるアプリケーションを追加・変更することができる。また、上記ファイルに対するアクセス条件の自由な設定により、本発明のファイルアクセス制御装置および制御方法は種々のソフトウェアに適用可能な汎用性が高いものとなる。

【0048】次にファイルアクセス制御部7におけるアクセス制御を具体的なアプリケーションの動作に沿って説明する。

【0049】図3はファイルアクセス制御部7による制御の一例を時系列的に示したものである。図3で例示するソフトウェアは、複数のアプリケーションソフトを含み、これらのアプリケーションソフトがユーザのコマンドによって一定の手順で次々に起動および実行されるように構成されているものとする。

【0050】図3ではメインのアプリケーションPRG-Aが起動され、このメインのアプリケーションPRG-AによってファイルFILE-Aが書き換えられる場合が示されている。

【0051】この場合、前記ファイル属性管理テーブル

2には、ユーザ α は直接メイン・アプリケーションPRG-Aを起動でき、メイン・アプリケーションPRG-Aを通じてサブ・アプリケーションPRG-Bを起動でき、サブ・アプリケーションPRG-Bを通じてはデータファイルFILE-Aを書き換えられるように予め設定しておくものとする。

【0052】図3において、コンピュータの立ち上げ後ユーザID α が獲得されたとすると、このユーザID α は正規ユーザのものであるか否かを判断され、正規ユーザのものである場合はユーザ情報格納部4に格納される(ステップ100)。

【0053】格納されたユーザID α は新たなユーザIDが獲得されない限り、すなわち、ソフトウェアを中止して再度コンピュータを立ち上げない限り有効に存続する。

【0054】なお、図3には示していないが、図1で説明したように、コンピュータの立ち上げ時に、アプリケーション情報管理部5は初期状態、すなわち何も登録されていない状態に戻される。

【0055】次にメイン・アプリケーションPRG-Aを実行するコマンドを入力したとすると、このPRG-Aに対するアクセスの情報がファイルアクセス制御部7によって検知され、ファイルアクセス制御部7は、PRG-Aを実行しようとするユーザとアプリケーションをそれぞれユーザ情報格納部4とアプリケーション情報管理部5によって検出する(ステップ110)。

【0056】この場合は、ユーザ情報格納部4からはユーザ情報 α を得るものの、アプリケーション情報管理部5が未登録の状態であるので、ユーザ α による直接のPRG-Aの読み出し命令であることがわかる。ここでは、ユーザ α によるメイン・アプリケーションPRG-Aの直接の読み出しは可能なように前記ファイル属性管理テーブル2に定義されているので、メイン・アプリケーションPRG-Aは実行され、同時に現在実行中のアプリケーションがPRG-Aであることがアプリケーション情報管理部5に登録される(ステップ120)。

【0057】上記アプリケーション情報管理部5に登録されたアプリケーション情報(PRG-A)は、PRG-Aの実行が終了するまでアプリケーション情報管理部5に有効に存続する。

【0058】次にソフトウェア実行中にPRG-Aによってサブ・アプリケーションPRG-Bの実行命令が発せられたとする。このサブ・アプリケーションPRG-Bに対するアクセス命令はファイルアクセス制御部7によって検知され、ファイルアクセス制御部7はサブ・アプリケーションPRG-Bに対する実行命令の発信源であるユーザとアプリケーションをユーザ情報格納部4とアプリケーション情報管理部5に問い合わせる(ステップ130)。

【0059】この結果、このPRG-Bの実行命令は、

ユーザ α がメイン・アプリケーションPRG-Aを通じて行っているものとわかり、ファイルアクセス制御部7はファイル属性管理テーブル2の設定条件に基づいてサブ・アプリケーションPRG-Bの実行命令を許可する。

【0060】サブ・アプリケーションPRG-Bが実行されると、ファイルアクセス制御部7は現在実行中のアプリケーションがサブ・アプリケーションPRG-Bであることをアプリケーション情報管理部5に追加登録する(ステップ140)。

【0061】次に、サブ・アプリケーションPRG-Bの実行によって、データファイルFILE-Aに対する書き込み命令が発せられたとする。ファイルアクセス制御部7は、このデータファイルFILE-Aに対する書き込み命令を発したユーザとアプリケーションをユーザ情報格納部4とアプリケーション情報管理部5へ問い合わせる。この結果、ユーザ α がサブ・アプリケーションPRG-Bを通じてデータファイルFILE-Aの内容を書き換えるものであると判断し、ファイル属性管理テーブル2の設定条件によってこの書き込み命令を禁止する(ステップ150)。

【0062】続いてサブ・アプリケーションPRG-Bの実行が終了した場合について説明する。サブ・アプリケーションPRG-Bの実行が終了すると、ファイルアクセス制御部7はこれを検知し、アプリケーション情報管理部5に登録されていたサブ・アプリケーションPRG-Bを消去する(ステップ160)。

【0063】サブ・アプリケーションPRG-Bの実行が終了すると、中断されていたメイン・アプリケーションPRG-Aが実行され、さらに、この状態でメイン・アプリケーションPRG-Aの実行によってデータファイルFILE-Aに対する書き込み命令が発せられたとする。この場合は、ファイルアクセス制御部7はユーザ α がメイン・アプリケーションPRG-Aを通じてデータファイルFILE-Aを書き換えようとしたものと判断し、ファイル属性管理テーブル2の設定条件によってデータファイルFILE-Aの書き換えを許可する(ステップ170)。

【0064】図4および図5は本発明によるファイルアクセス制御装置および制御方法による制御のパターンを例示したものである。

【0065】図4はユーザ管理を中心に、所定のユーザXが特定のプログラムAの動作によってのみファイル1を更新/消去できるように制御する場合を示している。

【0066】図4(a)は、ユーザXがプログラムAを通じて行うファイル処理を示しており、読み出し、書き込み、消去、実行のいずれも行うことができる。

【0067】これに対して、図4(b)はユーザXが指定以外のプログラムBを通じて行うファイル処理を示しており、この場合は、誤作動によるデータ破壊を防止す

10

20

30

40

50

12

【発明の効果】上記説明から明らかなように、本願請求項1および請求項3に係るファイルアクセス制御装置および制御方法は、ファイル属性管理テーブルと、アプリ

10

20

【符号の説明】

- 1 ファイルアクセス制御装置
- 2 ファイル属性管理テーブル
- 3 ファイル属性管理テーブル設定部
- 4 ユーザ情報格納部
- 5 アプリケーション情報管理部
- 6 アプリケーション情報初期設定部
- 7 ファイルアクセス制御部

【图2】

設定画面検索条件

ファイル名 _____ ユーザ名 _____ ユーザグループ名 _____ タイプ _____ アプリケーションID _____

				ユーザ管理			アプリケーション管理	
				ユーザ	ユーザグループ	ユーザ・ユーザグループ外	同一ID	同一ID外
名称	ユーザ	ユーザグループ	アプリケーションID	読書実行	読書実行	読書実行	読書実行	読書実行
main.exe	α	β	XXX	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
image.dat	α	β	XXX	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
sub.exe	α	β	XXX	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
hiscore.dat	α	β	XXX	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> <input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>

【図1】

